Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	33	(kelly near edmund).in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:57
L2	20	(cmelik near robert).in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:57
L3	17	(wing near malcolm).in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:57
L4	33	(kelly near edmund).in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L5	20	(cmelik near robert).in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L6	17	(wing near malcolm).in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L7	43	L4 or L5 or L6	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L8	563587	software	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L9	428695	hardware	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L10	277	DLAT	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58

L11	4699	TLB	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L12	1323157	target address	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L13	698810	host instruction	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L14	45435	emulat$4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L15	246269	L8 and L9	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L16	57	L15 and L10	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L17	15	L16 and L11	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L18	15	L17 and L12	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L19	12	L18 and L13	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58
L20	3	L19 and L14	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/08/30 15:58

acm PORTAL

USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+translation +lookaside +buffer, +TLB, +translation +consiste

SEARCH

[redacted]

Feedback Report a problem Satisfaction survey

Terms used
translation lookaside buffer TLB translation consistency address coherency

Found **15** of **160,457**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 15 of 15

Relevance scale □ ▭ ▬ ▬ ■

1 Translation lookaside buffer consistency: a software approach ■

D. L. Black, R. F. Rashid, D. B. Golub, C. R. Hill

April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the third international conference on Architectural support for programming languages and operating systems**, Volume 17 Issue 2

Full text available: pdf(1.38 MB) Additional Information: full citation, abstract, references, citings, index terms

We discuss the translation lookaside buffer (TLB) consistency problem for multiprocessors, and introduce the Mach shootdown algorithm for maintaining TLB consistency in software. This algorithm has been implemented on several multiprocessors, and is in regular production use. Performance evaluations establish the basic costs of the algorithm and show that it has minimal impact on application performance. As a result, TLB consistency does not pose an insurmountable obstacle to multiprocessor ...

2 An in-cache address translation mechanism ■

D. A. Wood, S. J. Eggers, G. Gibson, M. D. Hill, J. M. Pendleton

June 1986 **ACM SIGARCH Computer Architecture News , Proceedings of the 13th annual international symposium on Computer architecture**, Volume 14 Issue 2

Full text available: pdf(770.30 KB) Additional Information: full citation, abstract, references, citings, index terms

In the design of SPUR, a high-performance multiprocessor workstation, the use of large caches and hardware-supported cache consistency suggests a new approach to virtual address translation. By performing translation in each processor's virtually-tagged cache, the need for separate translation lookaside buffers (TLBs) is eliminated. Eliminating the TLB substantially reduces the hardware cost and complexity of the translation mechanism and eliminates the translation consistency problem. Trac ...

3 Options for dynamic address translation in COMAs ■

Xiaogang Qiu, Michel Dubois

April 1998 **ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture**, Volume 26 Issue 3

Full text available: pdf(1.37 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

In modern processors, the dynamic translation of virtual addresses to support virtual memory is done before or in parallel with the first-level cache access. As processor technology improves at a rapid pace and the working sets of new applications grow insatiably the latency and bandwidth demands on the TLB (Translation Lookaside Buffer) are getting more and more difficult to meet. The situation is worse in multiprocessor systems, which run larger applications and are plagued by the TLB consiste ...

4 Cache Memories
Alan Jay Smith
September 1982 **ACM Computing Surveys (CSUR)**, Volume 14 Issue 3
Full text available: pdf(4.61 MB) Additional Information: full citation, references, citings, index terms

5 Architecture support for single address space operating systems
Eric J. Koldinger, Jeffrey S. Chase, Susan J. Eggers
September 1992 **ACM SIGPLAN Notices , Proceedings of the fifth international conference on Architectural support for programming languages and operating systems**, Volume 27 Issue 9
Full text available: pdf(1.39 MB) Additional Information: full citation, references, citings, index terms

6 Coherency for multiprocessor virtual address caches
James R. Goodman
October 1987 **Proceedings of the second international conference on Architectual support for programming languages and operating systems**, Volume 15 , 22 , 21 Issue 5 , 10 , 4
Full text available: pdf(962.36 KB) Additional Information: full citation, abstract, references, citings, index terms

A multiprocessor cache memory system is described that supplies data to the processor based on virtual addresses, but maintains consistency in the main memory, both across caches and across virtual address spaces. Pages in the same or different address spaces may be mapped to share a single physical page. The same hardware is used for maintaining consistency both among caches and among virtual addresses. Three different notions of a cache "block" are defined: (1) the unit for transferring data t ...

7 SoftFLASH: analyzing the performance of clustered distributed virtual shared memory
Andrew Erlichson, Neal Nuckolls, Greg Chesson, John Hennessy
September 1996 **Proceedings of the seventh international conference on Architectural support for programming languages and operating systems**, Volume 31 , 30 Issue 9 , 5
Full text available: pdf(1.29 MB) Additional Information: full citation, abstract, references, citings, index terms

One potentially attractive way to build large-scale shared-memory machines is to use small-scale to medium-scale shared-memory machines as clusters that are interconnected with an off-the-shelf network. To create a shared-memory programming environment across the clusters, it is possible to use a virtual shared-memory software layer. Because of the low latency and high bandwidth of the interconnect available within each cluster, there are clear advantages in making the clusters as large as possi ...

8 The effects of virtually addressed caches on virtual memory design and performance
Jon Inouye, Ravindranath Konuru, Jonathan Walpole, Bart Sears
October 1992 **ACM SIGOPS Operating Systems Review**, Volume 26 Issue 4
Full text available: pdf(1.32 MB) Additional Information: full citation, abstract, index terms

Recent times have witnessed rapid advances in microprocessor technology resulting in an order of magnitude performance improvement every few years. These developments in hardware have been paralleled by several prominent trends in operating system design, the most notable being a move towards message-passing micro-kernels. However, operating system performance has not kept pace with that of the underlying hardware. It has become apparent that design changes to enhance processor performance can h ...

9 The Kyushu University reconfigurable parallel processor: design of memory and intercommunicaiton architectures
Kazuaki Murakami, Shin-ichiro Mori, Akira Fukuda, Toshinori Sueyoshi, Shinji Tomita
June 1986 **Proceedings of the 3rd international conference on Supercomputing**

Full text available: pdf(1.39 MB) Additional Information: full citation, abstract, references, citings, index terms

The reconfigurable parallel processor system under development at Kyushu University is an MIMD-type multiprocessor which consists of N processing-elements (currently N is 128) fully connected by S N × N crossbar networks (currently S is 1). Each PE (Processing Element) employs a Fujitsu SPARC MB86900/10 chip-set, a Weitek WTL1164/65 chip-set, an MMU (Memory Management Unit) with 64K b ...

10 The VMP multiprocessor: initial experience, refinements, and performance evaluation
D. R. Cheriton, A. Gupta, P. D. Boyle, H. A. Goosen
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture**, Volume 16 Issue 2

Full text available: pdf(1.73 MB) Additional Information: full citation, abstract, references, citings, index terms

VMP is an experimental multiprocessor being developed at Stanford University, suitable for high-performance workstations and server machines. Its primary novelty lies in the use of software management of the per-processor caches and the design decisions in the cache and bus that make this approach feasible. The design and some uniprocessor trace-driven simulations indicating its performance have been reported previously. In this paper, we present our initial experience with the V ...

11 Supporting reference and dirty bits in SPUR's virtual address cache
D. A. Wood, R. H. Katz
April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the 16th annual international symposium on Computer architecture**, Volume 17 Issue 3

Full text available: pdf(1.12 MB) Additional Information: full citation, abstract, references, citings, index terms

Virtual address caches can provide faster access times than physical address caches, because translation is only required on cache misses. However, because we don't check the translation information on each cache access, maintaining reference and dirty bits is more difficult. In this paper we examine the trade-offs in supporting reference and dirty bits in a virtual address cache. We use measurements from a uniprocessor SPUR prototype to evaluate different alternatives. The prototype's buil ...

12 Hardware-software trade-offs in a direct Rambus implementation of the RAMpage memory hierarchy
Philip Machanick, Pierre Salverda, Lance Pompe
October 1998 **Proceedings of the eighth international conference on Architectural support for programming languages and operating systems**, Volume 32 , 33 Issue 5 , 11

Full text available: pdf(1.47 MB) Additional Information: full citation, abstract, references, citings, index terms

The RAMpage memory hierarchy is an alternative to the traditional division between cache and main memory: main memory is moved up a level and DRAM is used as a paging device. The idea behind RAMpage is to reduce hardware complexity, if at the cost of software complexity, with a view to allowing more flexible memory system design. This paper investigates some issues in choosing between RAMpage and a conventionalcache architecture, with a view to illustrating trade-offs which can be made in choosi ...

13 Compiler transformations for high-performance computing
David F. Bacon, Susan L. Graham, Oliver J. Sharp
December 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 4

Full text available: pdf(6.32 MB) Additional Information: full citation, abstract, references, citings, index terms, review

In the last three decades a large number of compiler transformations for optimizing programs have been implemented. Most optimizations for uniprocessors reduce the number of instructions executed by the program using transformations based on the analysis of scalar quantities and data-flow techniques. In contrast, optimizations for high-performance superscalar, vector, and parallel processors maximize parallelism and memory locality with transformations that rely on tracking the properties o ...

Keywords: compilation, dependence analysis, locality, multiprocessors, optimization, parallelism, superscalar processors, vectorization

14 Limits to low-latency communication on high-speed networks
Chandramohan A. Thekkath, Henry M. Levy
May 1993 **ACM Transactions on Computer Systems (TOCS)**, Volume 11 Issue 2

Full text available: pdf(1.96 MB) Additional Information: full citation, abstract, references, citings, index terms, review

The throughput of local area networks is rapidly increasing. For example, the bandwidth of new ATM networks and FDDI token rings is an order of magnitude greater than that of Ethernets. Other network technologies promise a bandwidth increase of yet another order of magnitude in several years. However, in distributed systems, lowered latency rather than increased throughput is often of primary concern. This paper examines the system-level effects of newer high-speed network technologies on l ...

Keywords: ATM networks, host-network interfaces, interprocess communication, remote procedure calls, transport level protocols

15 A progress report on SPUR: February 1, 1987
Dave Patterson
March 1987 **ACM SIGARCH Computer Architecture News**, Volume 15 Issue 1

Full text available: pdf(408.51 KB) Additional Information: full citation, index terms

Results 1 - 15 of 15

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player